



①9 BUNDESREPUBLIK
DEUTSCHLAND



DEUTSCHES
PATENT- UND
MARKENAMT

⑫ **Offenl gungsschrift**
⑩ **DE 199 29 331 A 1**

⑤ Int. Cl.⁷:
B 60 R 16/02

⑲ Aktenzeichen: 199 29 331.7
⑳ Anmeldetag: 26. 6. 1999
㉑ Offenlegungstag: 4. 1. 2001

DE 199 29 331 A 1

⑦1 Anmelder:

DaimlerChrysler AG, 70567 Stuttgart, DE; Robert
Bosch GmbH, 70469 Stuttgart, DE

⑦2 Erfinder:

Göbel, Fridjof, Dipl.-Ing., 74080 Heilbronn, DE;
Schneider, Jutta, Dr., 70839 Gerlingen, DE; Hahn,
Ralf, 55252 Mainz-Kastel, DE; Hudel, Peter, 61169
Friedberg, DE; Koburg, Cornelius, 64319
Pfungstadt, DE; Lutz, Peter, 61479 Glashütten, DE

⑤6 Entgegenhaltungen:

DE 197 20 285 A1
DE 196 25 002 A1

NEUMANN, K.J. et. al.: "Ein aufkommender
Standort für vorteilte Systeme im Kfz", atp
4/98, S. 22;

Die folgenden Angaben sind den vom Anmelder eingereichten Unterlagen entnommen

Prüfungsantrag gem. § 44 PatG ist gestellt

⑤4 Vernetztes Fahrzeugkommunikationssystem mit Frontend-Einheit, benutzerbedienbarem Endgerät und zugehöriger Applikation

⑤7 Die Erfindung bezieht sich auf ein Fahrzeugkommunikationssystem mit einem Datenbus und wenigstens einer daran angeschlossenen Frontend-Einheit mit Benutzerschnittstellen-Frameworkeinheit, mit wenigstens einem benutzerbedienbaren, an dem Datenbus angeschlossenen Endgerät und mit wenigstens einer in das System implementierten, unter Beteiligung der Frontend-Einheit und des Endgerätes ausführbaren Funktionalität. Erfindungsgemäß ist die implementierte Funktionalität in einem mit der Benutzerschnittstellen-Frameworkeinheit kommunizierenden Teil mit Benutzerschnittstellenseiten und in einem mit dem Benutzerschnittstellenseitenteil einerseits und einer Applikations-Frameworkeinheit andererseits kommunizierenden Funktionskomponententeil aufgeteilt. Der Benutzerschnittstellenseitenteil befindet sich in der einen Frontend-Einheit, während sich der Funktionskomponententeil ebenfalls dort oder in einer anderen Frontend-Einheit oder in einer ebenfalls an den Datenbus angeschlossenen Mehrzweckplattform-Einheit befindet.
Verwendung z. B. in Automobilen.

DE 199 29 331 A 1

Die Erfindung bezieht sich auf ein Fahrzeugkommunikationssystem mit einem Datenbus, an den wenigstens eine Frontend-Einheit mit Benutzerschnittstellen-Frameworkeinheit und ein benutzerbedienbares Endgerät angeschlossen sind, sowie wenigstens einer in das System implementierten, unter Beteiligung der Frontend-Einheit und des Endgerätes ausführbaren Applikation, nachfolgend auch Funktionalität bezeichnet.

In modernen Kraftfahrzeugen, insbesondere auch Automobilen, sind zahlreiche Funktionalitäten implementiert, die unter Beteiligung jeweils zugehöriger Frontend-Einheiten, welche zugehörige Benutzerschnittstellen beinhalten, im Dialog mit dem Systemnutzer ausgeführt werden, insbesondere zahlreiche Telematik-Anwendungen, wie sie beispielsweise in der Offenlegungsschrift DE 196 25 002 A1 angeführt sind.

Um die sich daraus ergebenden Anforderungen bestmöglich zu erfüllen, wird in jüngerer Zeit zunehmend die Verwendung sogenannter verteilter Systeme betrachtet, insbesondere solchen, die auf objektorientierten Komponentenmodellen basieren. In der allgemeinen EDV mit den dort gegenüber Fahrzeuganwendungen vorhandenen höheren Rechenkapazitäten sind bereits Technologien zur Unterstützung solcher verteilter Komponentensysteme gebräuchlich, z. B. CORBA (Common Object Request Broker Architecture) und DCOM (Distributed Component Object Model). Diese Techniken lassen sich jedoch nicht gut auf kleine eingebettete Einheiten skalieren. Zudem ist DCOM jedenfalls gegenwärtig noch nicht für das Betriebssystem Windows CE erhältlich. Zu fahrzeugseitigen Konzepten für verteilte Systeme sei z. B. auf den Zeitschriftenaufsatz K. J. Neumann et al., ein aufkommender Standard für verteilte Systeme im Kfz, atp 4/98, Seite 22 und die ältere deutsche Patentanmeldung 199 09 157.9 nebst der dort zitierten Literatur hingewiesen.

Der Erfindung liegt als technisches Problem die Bereitstellung eines Fahrzeugkommunikationssystems, der eingangs genannten Art zugrunde, das mit für Fahrzeuganwendungen vertretbarem Rechenaufwand einen Mechanismus für die Realisierung von verteilten, auszuführenden Applikationen unter Nutzung von in Fahrzeugen gebräuchlichen Datenbusnetzwerken ermöglicht und bei dem die Applikation dabei möglichst unabhängig vom Typ des verwendeten Bussystems gehalten werden kann.

Die Erfindung löst dieses Problem durch die Bereitstellung eines Fahrzeugkommunikationssystems mit den Merkmalen des Anspruchs 1. Bei diesem System ist eine Aufteilung der jeweiligen implementierten Funktionalität in einen Teil mit Benutzerschnittstellenseiten und in einen Teil mit den Funktionskomponenten vorgesehen. Wenigstens der Benutzerschnittstellenseitenteil ist Teil der Frontend-Einheit, während sich der Funktionskomponententeil ebenfalls dort oder aber in einer anderen Frontend-Einheit oder einer ebenfalls an den Datenbus angeschlossenen Mehrzweckplattform-Einheit befindet. Der Benutzerschnittstellenseitenteil steht mit der Benutzerschnittstellen-Frameworkeinheit in der betreffenden Frontend-Einheit in Verbindung, während entsprechend der Funktionskomponententeil mit einer zugehörigen Applikations-Frameworkeinheit in Kommunikationsverbindung steht. Dieser Systemaufbau stellt einen Mechanismus für die Realisierung von verteilten, auszuführenden Funktionalitäten bereit, insbesondere auch hinsichtlich der dazu auszuführenden Teilfunktionen, wie Anzeige, Bedienung und Interaktion mit anderen Applikationsmodulen, mit einem für Fahrzeuganwendungen vertretbaren Rechenaufwand.

In Weiterbildung der Erfindung nach Anspruch 2 ist dem jeweiligen Funktionskomponententeil eine virtuelle Endgeräteinheit zugeordnet, über welche die Kommunikation mit dem betreffenden benutzerbedienbaren Endgerät über den Datenbus erfolgt und welche die hierzu benötigten busspezifischen Implementierungsinformationen enthält. Dadurch lassen sich die Funktionskomponenten im wesentlichen unabhängig vom Typ des jeweils verwendeten Datenbussystems halten, so daß sie nicht zwingend jedes Mal implementiert werden müssen, wenn ein anderer Bustyp zum Einsatz kommt.

Bei einem nach Anspruch 3 weitergebildeten System befindet sich der Funktionskomponententeil nicht in der gleichen Einheit wie der Benutzerschnittstellenseitenteil, sondern in einer anderen, an den Datenbus angeschlossenen Einheit. Die Kommunikation zwischen beiden Teilen erfolgt dann in Form einer Proxy-Stub-Kommunikation über den Datenbus. Bei dieser Systemauslegung brauchen die Benutzerschnittstellenseiten keine Kenntnis über die aktuell vorhandene, verteilte Systemumgebung besitzen, sondern sie greifen auf die zugeordneten Proxy-Komponenten zu, welche die notwendigen Netzwerkoperationen realisieren. In der anderen Einheit, d. h. einer weiteren Frontend-Einheit oder einer vorzugsweise zentral für mehrere Frontend-Einheiten vorgesehenen Mehrzweckplattform-Einheit, fungiert die jeweilige Stub-Komponente als Client der Funktionskomponenten und kommuniziert mit der zugehörigen Proxy-Komponente der erstgenannten Frontend-Einheit. Da der gesamte Netzwerkcode in den Proxy- und Stub-Komponenten sitzt, kann jeglicher applikationsspezifischer Code völlig unabhängig vom zugrundeliegenden Datenbusnetzwerk gehalten werden, ohne in die Vernetzung involviert zu sein, was seine Codierung und Aufrechterhaltung vereinfacht.

Vorteilhafte Ausführungsformen der Erfindung sind in den Zeichnungen dargestellt und werden nachfolgend beschrieben. Hierbei zeigen:

Fig. 1 ein schematisches, ausschnittweises Blockdiagramm eines Fahrzeugkommunikationssystems mit Frontend-Einheit und Mehrzweckplattform-Einheit,

Fig. 2 ein schematisches, ausschnittweises Blockdiagramm eines Fahrzeugkommunikationssystems mit gemeinsamer Implementierung von Benutzerschnittstellenseitenteil und Funktionskomponententeil einer Applikation in einer Frontend-Einheit und

Fig. 3 ein schematisches, ausschnittweises Blockdiagramm eines Fahrzeugkommunikationssystems mit zwei Frontend-Einheiten und auf diese verteilten Funktionalitäten.

Bei dem in **Fig. 1** gezeigten, ersten Ausführungsbeispiel beinhaltet das Fahrzeugkommunikationssystem einen herkömmlichen optischen Datenbus 1 des Typs MOST, an den eine Frontendeinheit 2, ein benutzerbedienbares Endgerät 3 und eine Mehrzweckplattform-Einheit 4 angeschlossen sind. Weitere Bestandteile des Fahrzeugkommunikationssystems, insbesondere noch weitere Frontend-Einheiten und benutzerbedienbare Endgeräte sowie Fahrzeugsteuergeräte, sind der Übersichtlichkeit halber nicht gezeigt. Im gezeigten Systemteil sind zur Erläuterung der Erfindung beispielhaft zwei Applikationen implementiert, nämlich eine Radio-Funktionalität und eine Adressbuch-Funktionalität. Bei der Frontend-Einheit 2 handelt es sich konkret um eine Anzeige- und Bedieneinheit 2 für diese Funktionalitäten, während es sich bei dem benutzerbedienbaren, zugehörigen Endgerät 3 und einem für den MOST-Datenbus 1 ausgelegten FM-Tuner handelt.

Charakteristischerweise ist die jeweilige Funktionalität jeweils in einen in die Frontend-Einheit 2 implementierten

Benutzerschnittstellenseitenteil und einen in die Mehrzweckplattform-Einheit 4 implementierten Funktionskomponententeil aufgeteilt, d. h. die Frontend-Einheit 2 beinhaltet eine Komponente 5 mit Benutzerschnittstellenseiten (UI pages) für die Radio-Funktionalität und eine Komponente 6 mit Benutzerschnittstellenseiten für die Adressbuch-Funktionalität. Entsprechend besitzt die Mehrzweckplattform (MPP)-Einheit 4 einen Funktionskomponententeil 7 für die Radio-Funktionalität und einen Funktionskomponententeil 8 für die Adressbuch-Funktionalität. Die Benutzerschnittstellenseitenteile 5, 6 stehen in der Frontend-Einheit 2 zum einen mit einer dort implementierten Benutzerschnittstellen-Frameworkeinheit 9 und zum anderen mit je einer zugehörigen Proxy-Komponente 10, 11 in Verbindung. In der MPP-Einheit 4 stehen die Funktionskomponententeile 7, 9 zum einen mit einer Applikations-Frameworkeinheit 12 und zum anderen mit zugehörigen Stub-Komponenten 13, 14 in Verbindung. Busseitig stehen die Proxy- und Stub-Komponenten 10, 11, 13, 14 mit einer für den gewählten Bustyp MOST geeigneten Netzwerkdienst-Endstufe (Most Net Services) 15, 16 in Verbindung, über welche die jeweilige Anbindung an den Datenbus 1 erfolgt.

In der MPP-Einheit ist zudem eine virtuelle Endgeräteeinheit 17 als virtuelles FM-Tunergerät implementiert. In dieser virtuellen Einheit ist die Kommunikation des zugehörigen Funktionskomponententeils, hier des Teils 7 für die Radio-Funktionalität, mit dem speziell verwendeten, benutzerbedienbaren Endgerät, hier dem für den MOST-Datenbus 1 bestimmten FM-Tuner 3, enthalten, d. h. die virtuelle Einheit 17 bewerkstelligt die Kommunikation ihres zugehörigen Funktionskomponententeils 7 mit dem speziellen zugehörigen benutzerbedienbaren Endgerät 3 über den vorhandenen Datenbus 1 und enthält dazu die busspezifischen Implementierungsdetails der Funktionskomponenten. Dies hat die erwünschte Konsequenz, daß die im Funktionskomponententeil 7 enthaltenen Funktionskomponenten unabhängig vom jeweils gerade benutzten Bussystem 1 bleiben und daher nicht automatisch mit einer Änderung desselben geändert werden müssen.

Bekanntermaßen bringt eine verteilte Systemauslegung eine Menge an Komplexität mit sich, wobei es empfehlenswert ist, die Systeme auf der höchsten verfügbaren Schicht im ISO/OSI-Netzwerk-Schichtenmodell mit den Schichten 1 (physikalische Verbindung) bis 7 (Präsentationsschicht) aufzubauen, um auf diese Weise so viele Netzwerkdetails wie möglich unsichtbar zu lassen und sich auf die Applikationsanforderungen des Systems zu konzentrieren. Genau dies beabsichtigt der MOST-Standard für Multimedia-Applikationen im Fahrzeug. MOST beschäftigt sich jedoch hauptsächlich mit der Kommunikation zwischen Endgeräten wie CD-Player, Tunern und CD-Wechslern und nicht mit der zwischen Frontend-Einheit 2 und MPP-Einheit 4 wünschenswerten Kommunikation auf höherem Level.

Die verwendete Proxy-Stub-Kommunikation ermöglicht die Erfüllung dieser Anforderungen unter Beibehaltung des MOST-Datenbusses 1, wobei die hierdurch gegebene, vorliegende Systemlösung als Prototyp COMOM bezeichnet ist, siehe das Systembeschreibungs-Insert in Fig. 1. Diese Systemlösung stellt eine Kommunikationsfunktion auf hohem Level zur Verfügung, analog zum COM/DCOM-Standard, und verwendet dabei die Steuerungs- und asynchronen Kanäle des MOST-Datenbusses 1. Da die Schnittstellennotation des Prototyps COMOM mit COM-IDL kompatibel ist, kann COMOM durch DCOM ersetzt werden, sobald diese Technologie für Windows CE verfügbar ist. Als weitere Alternative kommt der Einsatz von CORBA in Betracht, wenn ein anderes Betriebssystem verwendet wird. CORBA-Implementierungen sind für wichtige eingebettete

Betriebssysteme verfügbar, wie QNX, Chorus, VxWorks u. a.

Gemäß der Systemauslegung von Fig. 1 besteht der Benutzerschnittstellenteil der Applikation aus den Benutzerschnittstellenseiten 5, 6, welche in die Benutzerschnittstellen-Framework-Einheit 9 eingebettet sind, und mit dieser kommunizieren. Um auf die im jeweiligen Funktionskomponententeil 7, 8 implementierte Funktionalität zurückzugreifen, benötigen die Benutzerschnittstellenseiten 5, 8 keine Kenntnis über die aktuelle verteilte Systemumgebung. Stattdessen greifen sie auf die zugehörige Proxy-Komponente 10, 11 der Funktionskomponenten zu, die entsprechend agieren und dabei die notwendigen Netzwerkoperationen realisieren. Die in der MPP-Einheit 4 befindlichen Stub-Komponenten 13, 14 agieren als Clients der Funktionskomponenten 7, 8 und kommunizieren über den Datenbus 1 mit den Proxy-Komponenten 10, 11 auf Seiten der Frontend-Einheit 2. In den Proxy- und Stub-Komponenten 10, 11, 13, 14 ist der gesuchte Netzwerkcode abgelegt, so daß der gesamte applikationsspezifische Code völlig unabhängig vom zugrundeliegenden Datenbusnetzwerk 1 gehalten werden kann und nicht in die Vernetzung involviert sein braucht, was es leichter macht, ihn zu kodieren und beizubehalten. Die Proxy- und Stub-Komponenten 10, 11, 13, 14 können automatisch durch IDL-Compiler erzeugt werden, die für COM/DCOM- und CORBA-Produkte verfügbar sind.

Wie aus der obigen Beschreibung erkennbar, läßt sich das Fahrzeugkommunikationssystem erfindungsgemäß so auslegen, daß völlige Transparenz der Verteilung über das Netzwerk für die Komponenten erzielt wird. Während bei einer Auslegung von Fig. 1 die MPP-Einheit 4 vorzugsweise als zentrale Recheneinheit fungiert, die mehrere Frontend-Einheiten und mehrere Applikationen bedient, besteht ein weiterer Vorteil der Erfindung darin, daß auch eine Systemauslegung ohne Vernetzung möglich ist, d. h. ein System ohne die Funktionalität der MPP-Einheit 4 von Fig. 1. Ein solches System ist als zweites Ausführungsbeispiel der Erfindung in Fig. 2 ausschnittsweise dargestellt, wobei funktionell gleiche Bestandteile mit denselben Bezugszeichen wie in Fig. 1 versehen sind und insoweit auf die obige Beschreibung zu Fig. 1 verwiesen werden kann.

Wie aus Fig. 2 ersichtlich, sind bei dieser Systemrealisierung die Bestandteile der MPP-Einheit von Fig. 1 zusätzlich zu den Bestandteilen der Frontend-Einheit 2 von Fig. 1 in einer einzigen, insoweit modifizierten Frontend-Einheit 2a aufgenommen. Dadurch wird kein Netzwerkcode für die Kommunikation zwischen dem Benutzerschnittstellenseitenteil 5, 6 und dem zugehörigen Funktionskomponententeil 7, 8 einer jeweiligen Applikation benötigt, so daß hier die Proxy- und Stub-Komponenten 10, 11, 13, 14 des Systems von Fig. 1 entfallen. Die Benutzerschnittstellenseiten 5, 6 kommunizieren in diesem Beispiel direkt mit den zugehörigen Funktionskomponenten 7, 8. Diese Kommunikation kann z. B. wiederum über COM/CORBA realisiert sein, siehe auch das Systembeschreibungs-Insert von Fig. 2.

Eine weitere mögliche Systemauslegung ist in Fig. 3 dargestellt, wobei wiederum funktionell gleiche Bestandteile mit denselben Bezugszeichen wie in den Fig. 1 und 2 versehen sind und insoweit auf die obige Beschreibung zu den Fig. 1 und 2 verwiesen werden kann.

Wie aus Fig. 3 ersichtlich, sind die beiden beispielhaften Applikationen (Radio, Adressbuch) in dieser Realisierung in zwei Frontend-Einheiten 2b, 2c implementiert, von denen die eine Frontend-Einheit 2b analog zur Frontend-Einheit 2 von Fig. 2 ohne den Funktionskomponententeil aufgebaut ist, während die andere Frontend-Einheit 2c als selbständige Einheit entsprechend der Frontend-Einheit 2a von Fig. 2

ausgelegt ist, welche sowohl den jeweiligen Benutzerschnittstellenseitenteil 5, 6 als auch den zugehörigen Funktionskomponententeil 7, 8 enthält. Dabei genügt dann für die nicht mit dem Funktionskomponententeil versehene Frontend-Einheit 2b die Verwendung von weniger leistungsfähiger Hardware als für die andere Frontend-Einheit 2c, auf der die Funktionskomponenten ablaufen.

Nachfolgend werden beispielhaft einige mögliche Applikationen angegeben, die zugehörige, implementierte Software-Pakete umfassen. So kann ein Audio-Paket vorgesehen sein, das vorhandene Audio-Hardware steuert und die Führung der entsprechenden Daten von Audio-Quellen zu Audio-Ausgabegeräten vornimmt. Die zugehörige Audio-Hardware umfaßt eine zentrale Audio-Ausgabe über Lautsprecher und die Ausgabe über an Frontend-Einheiten verfügbare Kopfhörer unter Steuerung der Lautstärke und anderer Audio-Parameter. Dabei können fahrzeugspezifische Eigenschaften realisiert sein, wie z. B. eine Leiseschaltung einer gerade aktiven Audio-Quelle auf eine geringere Lautstärke, wenn eine Sprachausgabe aktiv wird.

Eine Telefon-Applikation kann zur Steuerung einer GSM-Funktionalität, zur Implementierung der TAPI, insbesondere für Sprachverbindungen, zum Wählen von Telefonnummern, Akzeptieren oder Zurückweisen von Anrufen und Verwalten einer Anrufliste, sowie zur Bereitstellung von Datenverbindungen über WAP dienen. Ein Radio-Paket ist, wie oben beschrieben, zur Steuerung der Tuner-Hardware, wie eines MOST-FM-Tuners, vorgesehen und ist zweckmäßigerweise so ausgelegt, daß es decodierte RDS-Nachrichten für andere Dienste, wie TMC, bereitstellt und Voreinstellungen sowie Stationslisten für die installierten Frontend-Einheiten verwaltet. Ein CD-Player/Video-Player-Paket steuert CD-Player-Hardware, die aus mehreren Geräten bestehen kann, z. B. einem CD-Wechsler und je einem CD-Player in mehreren Frontend-Einheiten. Das Paket übernimmt die Verwaltung von Abspiellisten, die Erzeugung beliebiger Abspiellisten und die optische Anzeige des momentanen Zustands der Abspielgeräte. Soweit vorhanden, kann dieses Paket auch zum Abspielen von Videodisks verantwortlich sein. Ein Fahrzeug-Paket kann dazu dienen, auf Fahrzeugdatenbussen, wie einem CAN-Bus, verfügbare Fahrzeuginformationen bereitzustellen, indem diese gesammelt und über vorgegebene Schnittstellen zur Verfügung gestellt werden. Bei diesen Fahrzeuginformationen kann es sich z. B. um solche über die Geschwindigkeit, die bevorratete Kraftstoffmenge und diverse Temperaturangaben handeln. Die Informationen können optisch angezeigt und/oder anderen Diensten verfügbar gemacht werden.

Ein Profil-Paket ist für die Speicherung von Profilen von Komponenten, zur Feststellung von Komponenten, die von einer Profiländerung betroffen sind, um eine Reinitialisierung vorzunehmen, und zur Erzeugung neuer Profile dienlich. Die Profile werden zur Speicherung von Einstellungen von Komponenten und der Benutzerschnittstelle verwendet, wie Farben, Audio-Einstellungen etc., und in jeder Frontend-Einheit separat abgelegt. Ein Personenmanagement-Paket enthält Funktionalität für persönliche Informationen über einen identifizierten Systemnutzer, wie ein persönliches Adressbuch, einen Kalender, eine Email-Funktion und Datensynchronisation mit anderen Datenbanken, wie anderen Fahrzeugen oder einem stationären PC-System, z. B. über das Internet oder mit Bluetooth-Technologie.

Ein weiteres Paket dient dazu, die Erlaubnis für angeforderte Benutzeraktionen zu erteilen und dabei insbesondere Konflikte zu behandeln, wenn mehrere Systemnutzer dieselbe Systemfunktionalität anfordern, diese zum jeweiligen Zeitpunkt jedoch nur von einem Nutzer benutzt werden kann. Dieses Paket kann außerdem dazu verwendet werden,

sitzplatzbezogene Nutzungsrechte für bestimmte Nutzer zu beschränken, z. B. für Kinder, bzw. die Profile für Freigaben und Anpassungen von Rechten zu verwalten. Ein Nutzeridentifikations-Paket ist für die Zuführung einer Nutzererkennung an Komponenten verantwortlich, die eine solche benötigen, wie Adressbuch, Email, Kalender etc. Der Systemnutzer muß dann seine Identität geeignet eingeben, z. B. mittels PIN, Paßwort, Fingerabdruck oder Netzhautabtastung. Ein Komfort-Paket ermöglicht die Steuerung von Klimatisierungseinheiten und anderen Komfort-Funktionen, soweit im Fahrzeug vorhanden.

Ein Telematikdienste-Paket ist verantwortlich für das Sammeln generischer Telematikdienstinformationen und das Verteilen an anfordernde Komponenten. Ein Aktualisierungs-Paket verwaltet Aktualisierungen und Aufrüstungen von Komponenten und ist verantwortlich für das Herunterfahren des Systems in einen Zustand, in welchem eine Aktualisierung/Aufrüstung durchgeführt werden kann, sowie für das erneute Verbringen des Systems in den normalen Betriebsmodus. Ein Initialisierungs-, Überwachungs- und Diagnose-Paket ist verantwortlich für die Durchführung von Start- und Initialisierungsvorgängen, enthält die Fähigkeit zur Überwachung des Betriebs von Software- und Hardware-Komponenten und unterstützt eine Diagnose des Systems.

Ein Zusatzdienste-Paket kann zur Bereitstellung von Funktionalitäten vorgesehen sein, die auf herkömmliche Systemfähigkeiten aufsetzen. Bei diesen Zusatzfunktionalitäten kann es sich z. B. um einen Notruf, z. B. automatisch bei einem erkannten schwereren Unfall oder halbautomatisch auf Benutzeranforderung jeweils in Form von übertragenen Daten über die Fahrzeugposition oder in Sprachform, eine Logbuch-Funktion, wie sie zur Kostenerfassung bei Firmenfahrzeugen und zu anderen Zwecken nützlich ist, sowie einen Diebstahlschutz handeln, mit dem das System bei Vorhandensein eines Ortungsmoduls eine Bewegung des Fahrzeugs außerhalb bestimmter Grenzen feststellen und den Fahrzeugbetrieb bei Bedarf stoppen kann.

Patentsprüche

1. Fahrzeugkommunikationssystem mit
 - einem Datenbus (1) und wenigstens einer daran angeschlossenen Frontendeinheit (2) mit einer Benutzerschnittstellen-Frameworkeinheit (9),
 - wenigstens einem benutzerbedienbaren, an den Datenbus angeschlossenen Endgerät (3), und
 - wenigstens einer in das System implementierten, unter Beteiligung der Frontend-Einheit und des Endgerätes ausführbaren Funktionalität, **dadurch gekennzeichnet**, daß
 - die implementierte Funktionalität in einen mit der Benutzerschnittstellen-Frameworkeinheit (9) kommunizierenden Teil (5, 6) mit Benutzerschnittstellenseiten und in einen mit dem Benutzerschnittstellenseitenteil einerseits und einer Applikations-Frameworkeinheit (12) andererseits kommunizierenden Funktionskomponententeil (7, 8) aufgeteilt ist und
 - sich der Benutzerschnittstellenseitenteil in der Frontend-Einheit (2) befindet und sich der Funktionskomponententeil (7, 8) ebenfalls in dieser Frontend-Einheit oder in einer anderen, an den Datenbus angeschlossenen Frontend-Einheit oder einer an den Datenbus angeschlossenen Mehrzweckplattform-Einheit (4) befindet.
2. Fahrzeugkommunikationssystem nach Anspruch 1, weiter dadurch gekennzeichnet, daß dem Funktions-

komponententeil (7) eine virtuelle Endgeräteeinheit (17) zugeordnet ist, über welche der Funktionskomponententeil zur Kommunikation mit dem benutzerbedienbaren Endgerät (3) über den Datenbus (1) verbunden ist und welche busspezifische Implementierungsinformationen enthält. 5

3. Fahrzeugkommunikationssystem nach Anspruch 1 oder 2, weiter dadurch gekennzeichnet, daß

- sich der Funktionskomponententeil (7, 8) in der anderen Frontend-Einheit (2c) oder der Mehrzweckplattform-Einheit (4) befindet und 10
- dem Funktionskomponententeil (7, 8) eine Stub-Komponente (13, 14) sowie dem Benutzerschnittstellenseitenteil (5, 6) eine Proxy-Komponente (10, 11) zur Kommunikation zwischen 15 Funktionskomponententeil und Benutzerschnittstellenseitenteil zugeordnet ist.

Hierzu 3 Seite(n) Zeichnungen

20

25

30

35

40

45

50

55

60

65

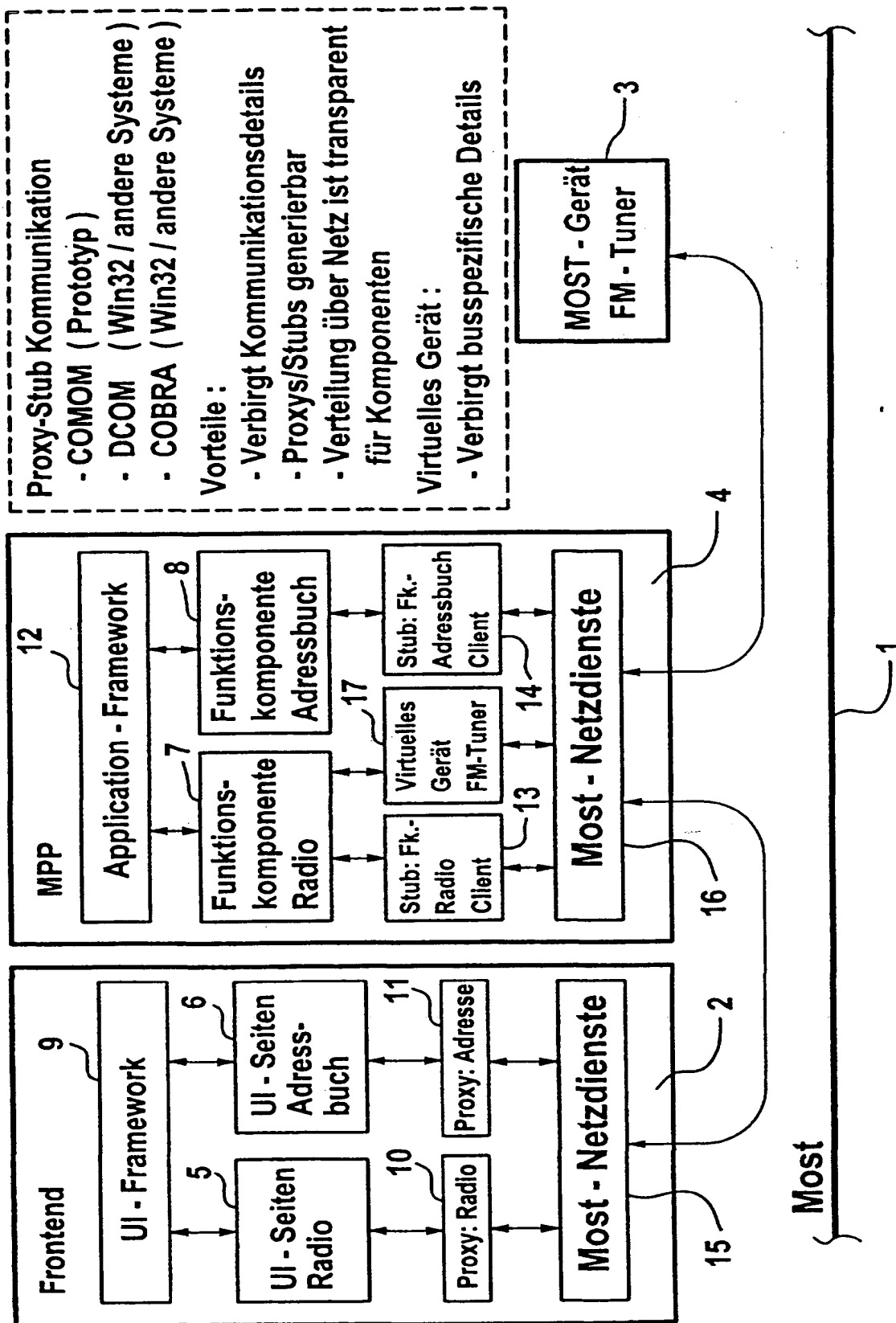


Fig. 1

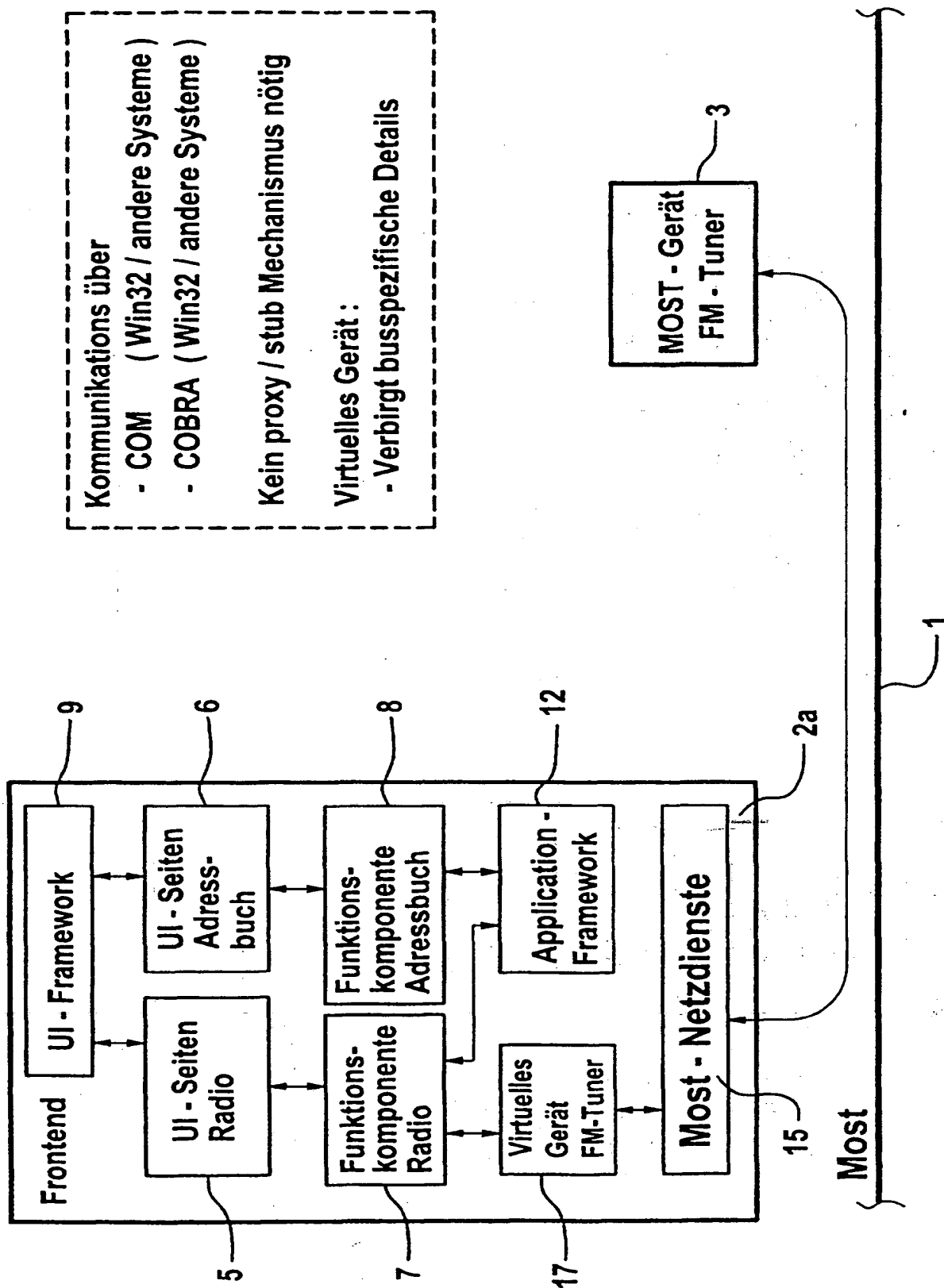


Fig. 2

Kommunikations über

- COM (Win32 / andere Systeme)
- COBRA (Win32 / andere Systeme)

Kein proxy / stub Mechanismus nötig

Virtuelles Gerät :

- Verbirgt busspezifische Details

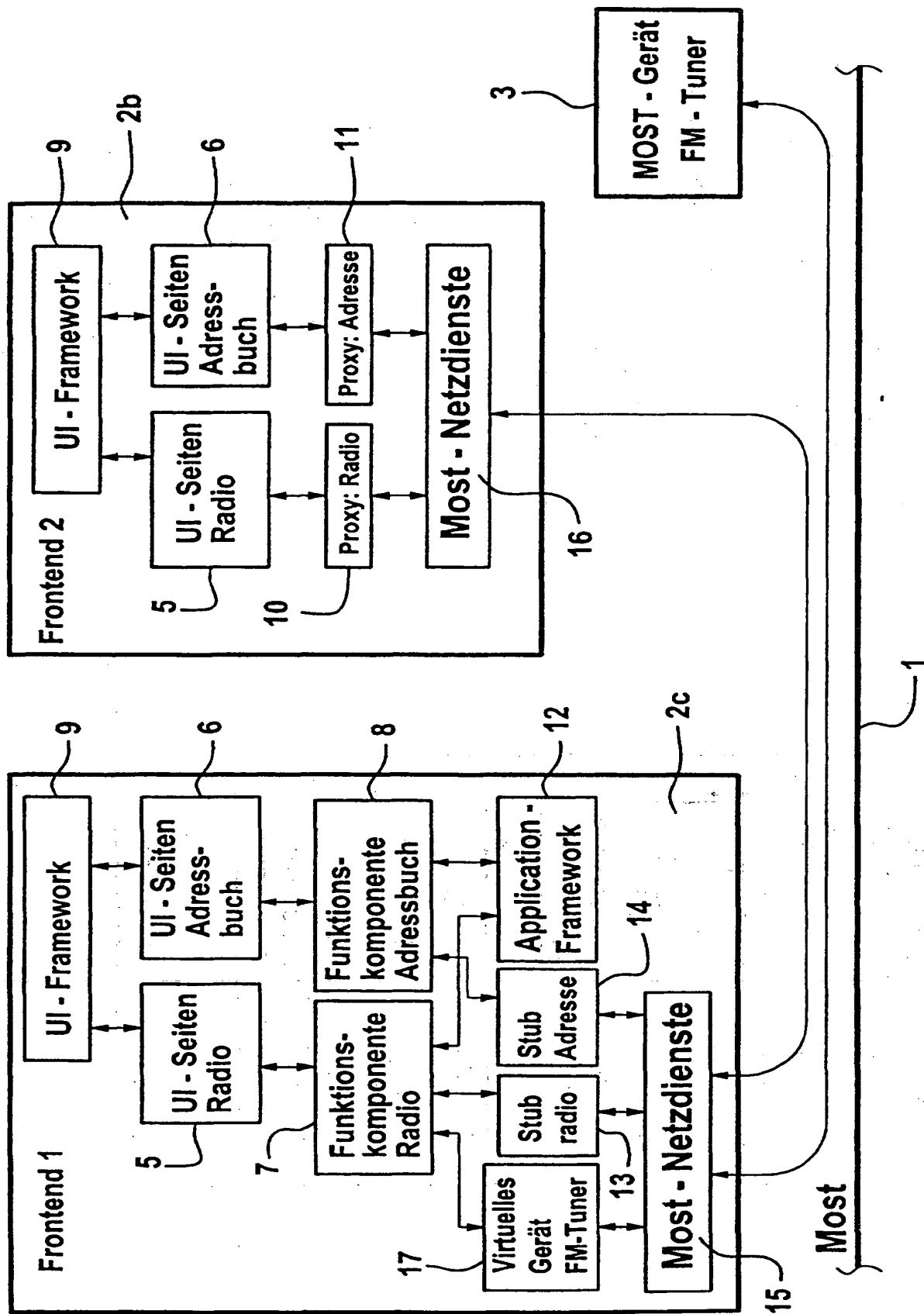


Fig. 3